

Й.Я. РИВКІНД, Т.І. ЛИСЕНКО,
Л.А. ЧЕРНІКОВА, В.В. ШАКОТЬКО

ІНФОРМАТИКА

8

Опрацювання текстових величин мовою програмування

За навчальною програмою 2017 року



teach-inf.com.ua



У ході виконання проєктів можна вводити дані та виводити результати.

Для введення даних використовується **поле, а для виведення результатів можна використовувати:**

поле

напис

вікно повідомлень





Створимо для прикладу проект, у якому обчислюватиметься сума двох довільних чисел.

Побудуємо математичну модель для цього проекту:

вхідні дані:

два довільних числа x і y ;

кінцеві результати:

їх сума z

формули:

$z = x + y$



Тому використаємо в цьому проєкті:

два поля

у які користувач під час виконання проєкту вводитиме доданки (вони можуть бути як цілі, так і дробові)

напис

для виведення значення суми

кнопку

після вибору якої відбудуватиметься додавання введених у поля чисел і виведення суми в напис



Розмістимо у вікні проекту два поля, напис і кнопку з текстом Сума. Проект, що створює вікно з розміщеними в ньому об'єктами, процедура для додавання двох чисел, а також вікно виконання проекту наведені на малюнку.

```
from tkinter import*
```

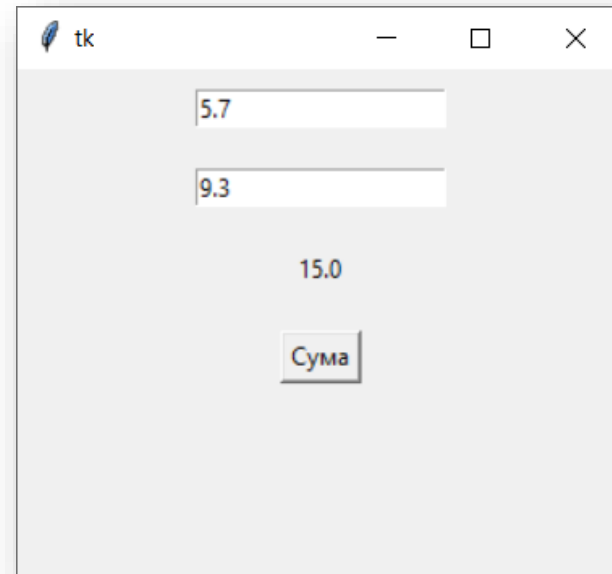
```
def click():  
    x = float(entry1.get())  
    y = float(entry2.get())  
    z = x + y  
    label['text'] = str(z)
```

```
root = Tk()  
root.geometry("300x250")
```

```
entry1 = Entry()  
entry1.pack(pady = 10)  
entry2 = Entry()  
entry2.pack(pady = 10)
```

```
label = Label()  
label.pack(pady = 10)
```

```
button = Button(text = 'Сума', command=click)  
button.pack(pady = 10)
```



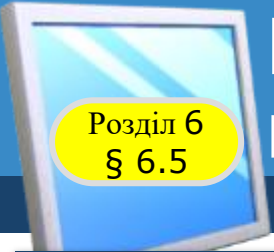


Розглянемо наведену процедуру.

Хоча користувач під час виконання проєкту вводить у поля числа, система тим не менше сприймає їх як тексти. Ці тексти є вмістом полів з іменами **entry1 і **entry2**. Тому перші дві команди процедури:**

```
x = float(entry1.get())  
y = float(entry2.get())
```

переводять текстове подання чисел безпосередньо в числа і присвоюють ці значення змінним **x і **y** відповідно.**



Проекти з уведенням даних і виведенням результатів



Третя команда додає ці числа та присвоює суму змінній **z**:

```
z = x + y
```

Четверта команда:

```
label['text'] = str(z)
```

(англ, *string* – рядок) переводить обчислену суму із числа в його текстове подання та присвоює цей текст значенню властивості **text** напису з іменем **label**.



У мові програмування *Python* змінним відповідають значення певного типу.

Тип визначає:

➤ **які значення може набувати відповідна змінна;**

➤ **які операції над значеннями можна виконувати;**

➤ **який обсяг оперативної пам'яті комп'ютера буде виділено для зберігання даних цього типу.**



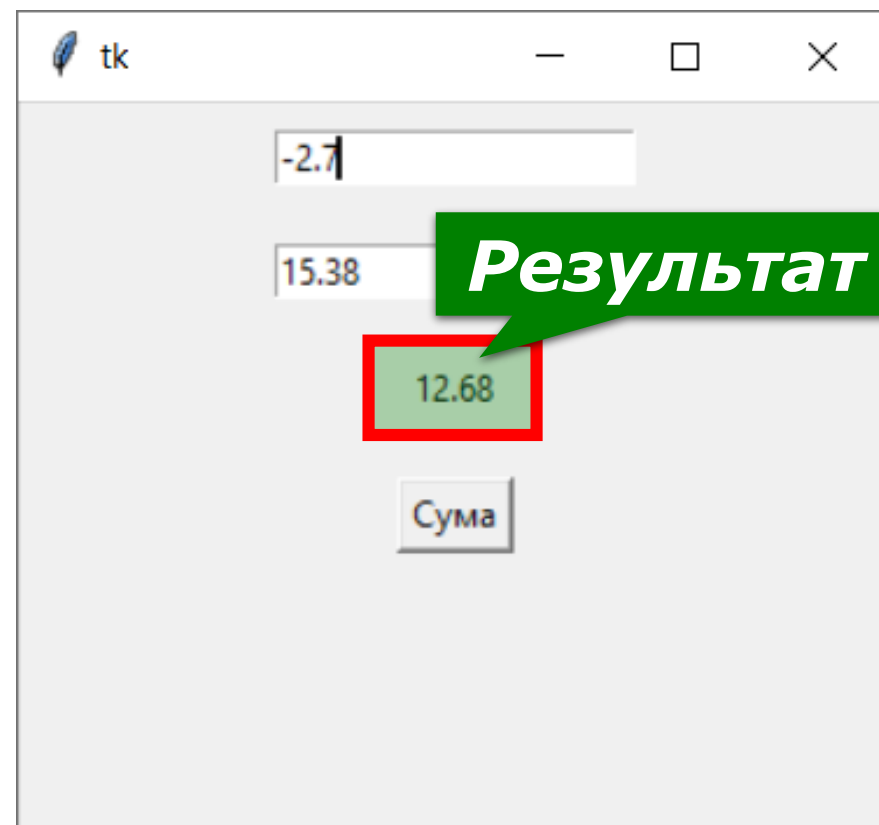


Значення змінних:

x**y****z**

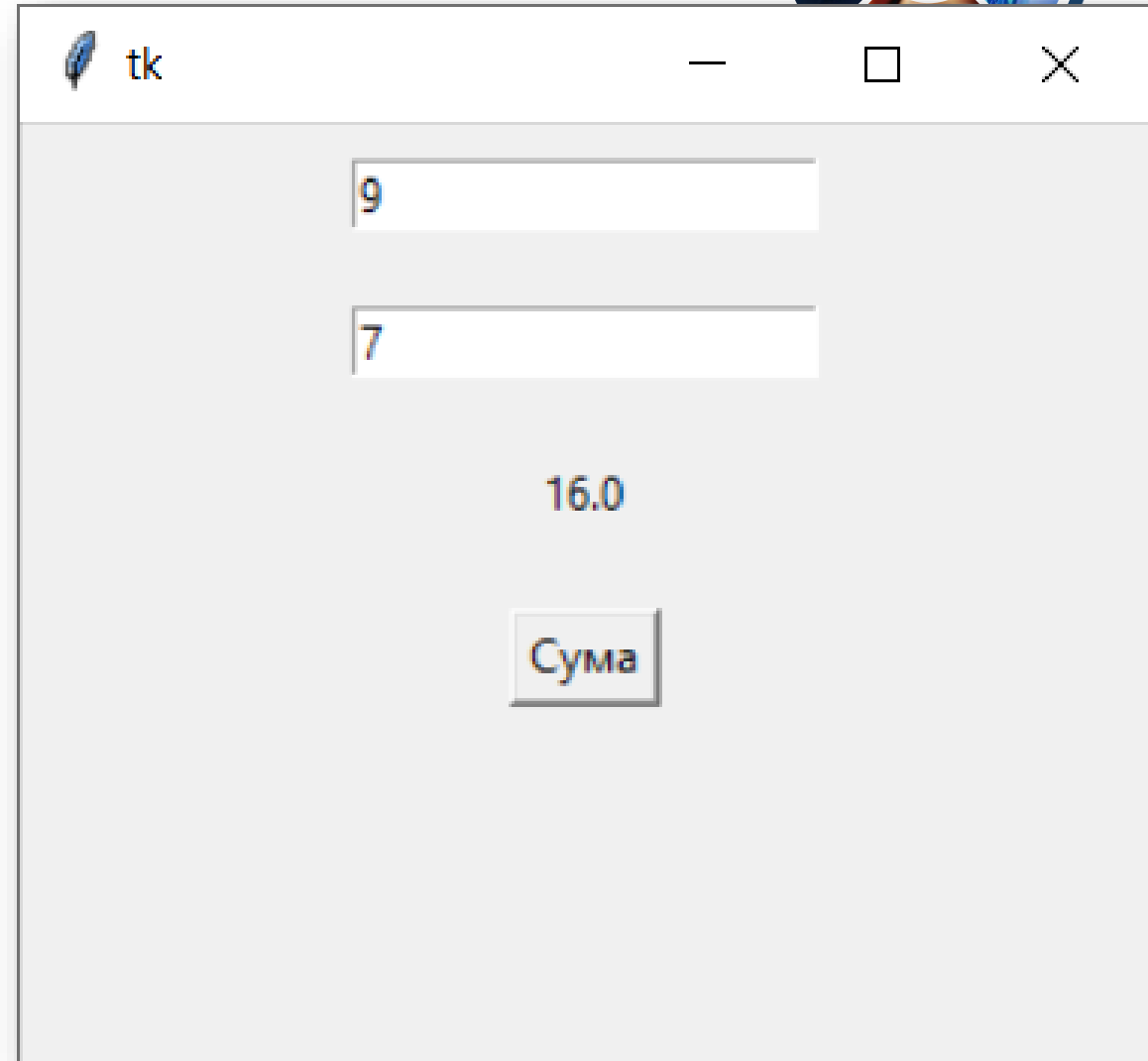
*Які використані в процедурі, за умовою задачі можуть бути довільними числами, тобто і цілими, і дробовими. Для таких значень можна використати тип **float**.*

*Під час виконання створеного проекту користувач уводить у поля два числа, вибирає кнопку із заголовком **Сума**, і результат виводиться в напис.*





Не закриваючи вікно виконання проєкту, користувач може ввести в поля інші числа, знову вибрати кнопку із заголовком **Сума** й одержати новий результат. І так можна повторювати довільну кількість разів.



Проекти з уведенням даних і виведенням результатів

Розділ 6
§ 6.5



*Щоб під час виконання проекту для користувача було зрозуміліше, у які поля що саме потрібно вводити і де виводитиметься результат, доцільно у вікні розмістити поруч з кожним полем і написом, призначеним для виведення результату, написи з пояснювальними текстами, наприклад **Перший доданок**, **Другий доданок**, **Сума**.*

tk

Перший доданок

5.17

Другий доданок

12.3

Сума

17.47

Сума



Якщо змінні можуть набувати тільки цілі значення, то для них можна використовувати тип **int** (англ. *integer-цілий*).

Якщо в проєкті використовуватимуться змінні типу **int**, для читання їх значень з поля можна використати, наприклад, команду:

```
x = int(entry1.get())
```



*У мові **Python** задачі з уведенням даних і виведенням результатів можуть бути розв'язані без створення графічного інтерфейсу користувача.*

Для введення даних під час виконання проєкту в команді присвоювання використовують функцію:

```
input('<текст>')
```



Наприклад, після виконання команди:

```
x = float(input('Уведіть перший доданок: '))
```

- у вікно середовища **IDLE** буде виведено текст, що міститься в дужках у лапках;
- користувачу потрібно ввести значення першого доданка з клавіатури та натиснути клавішу **Enter**;
- уведене значення першого доданка (цілого або дробового) буде сприйнято як текст, функцією **float** буде переведено в число, і це число присвоюється змінній **x**.



Якщо число, яке потрібно ввести, може бути тільки цілим, то потрібно замість **float** використати **int**.

Для виведення значень використовують команду:

```
print(<значення>)
```

Наприклад, команда:

```
print(z)
```

виведе значення змінної **z**;

```
print('Сума дорівнює')
```

виведе текст **Сума дорівнює**;

```
print('Сума дорівнює', z)
```

виведе текст **Сума дорівнює**, а після нього через пропуск значення змінної **z**.



Проект для розв'язування вищенаведеної задачі та результат його виконання в середовищі **IDLE наведені на малюнку.**

```
x = float(input('Уведіть перший доданок: '))  
y = float(input('Уведіть другий доданок: '))  
z = x + y  
print('Сума дорівнює', z)
```

```
Уведіть перший доданок: 12.3  
Уведіть другий доданок: 7.25  
Сума дорівнює 19.55
```



Ви вже знаєте, що в про'ктах в арифметичних виразах використовуються такі знаки арифметичних операцій:

+

ДОДАВАННЯ

-

ВІДНІМАННЯ

МНОЖЕННЯ

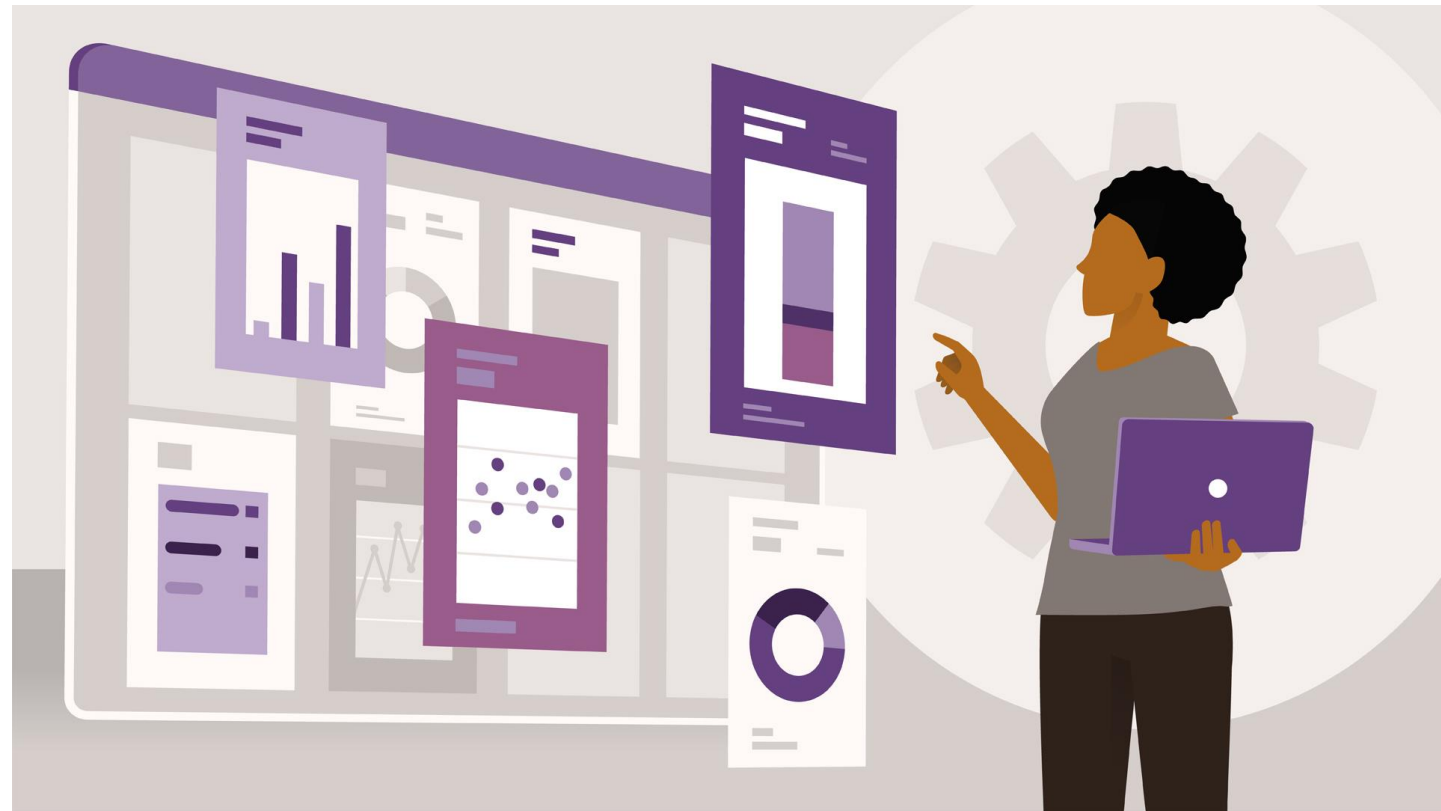
/

ДІЛЕННЯ



*Тип результату обчислення значень арифметичних виразів у мові **Python** залежить від типу величин, що є операндами в цьому виразі.*

*Так, якщо обидва операнди є величинами **цілого типу**, то в результаті їх додавання, віднімання та множення отримаємо значення **цілого типу**.*





Якщо хоча б один з операндів **дійсного типу**, то і результат виконання названих операцій буде також **дійсного типу**.

Результат ділення величин як **цілого**, так і **дійсного типу** завжди є **дійсним числом**.





*У проєкті, крім змінних, можна використовувати **константи** (постійні величини).*

Створимо проєкт, у якому потрібно обчислити масу сосни, якщо відомий її об'єм. Він може бути важливим під час вирішення питання, яка вантажівка потрібна для перевезення сосен на деревообробний завод.





Побудуємо математичну модель для цього проєкту:

вхідні дані:

об'єм сосни V

кінцеві результати:

маса сосни m

формули:

$$m = \rho * V$$

Де ρ – густина сосни – константа, що дорівнює 520 кг/м^3 .

Використання в проєкті констант (постійних величин)

Розділ 6
§ 6.5



*Вважатимемо, що в проєкті значення об'єму вводиться в кубічних метрах у поле з іменем **entry**, а значення маси виводиться в кілограмах у напис з іменем **label**.*

*У мові **Python** не можна створити сталі величини (константи), як в інших мовах програмування, щоб для них було заборонено змінювати значення величини.*

Натомість серед програмістів є домовленість записувати імена величин, значення яких не рекомендується змінювати, великими літерами.

A

Використання в проєкті констант (постійних величин)

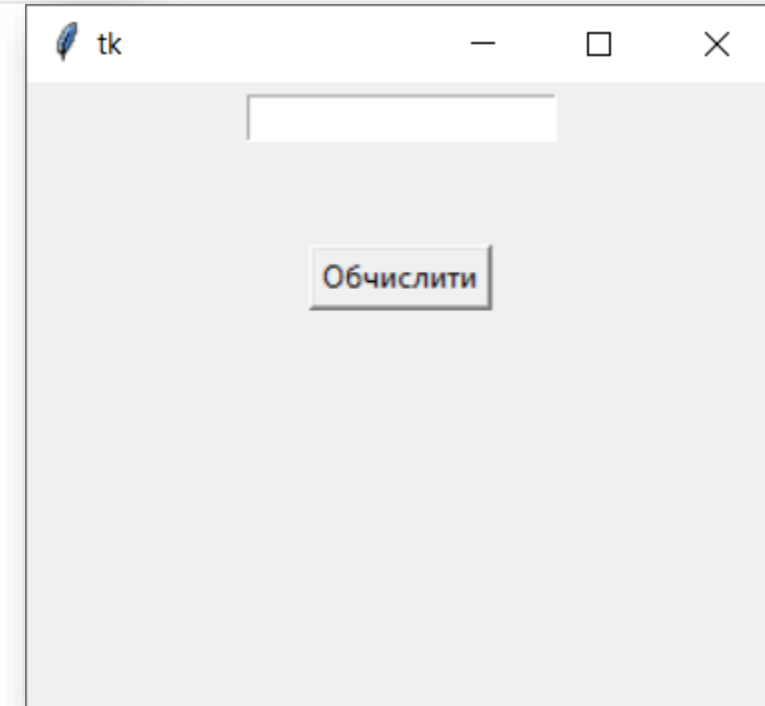
Розділ 6
§ 6.5



Тоді відповідний фрагмент проєкту виглядатиме так:

```
from tkinter import *
def click():
    P = 520
    v = float(entry.get())
    m = P * v
    label['text'] = str(m)

root = Tk()
root.geometry("300x250")
entry = Entry()
entry.pack(pady = 5)
label = Label()
label.pack(pady = 5)
button = Button(text = 'Обчислити', command=click)
button.pack(pady = 5)
```



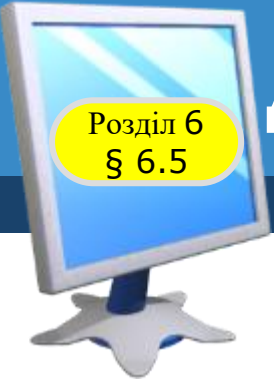


1. Якими командами можна перевести текстове подання числа безпосередньо в число й навпаки?

2. Що визначає тип змінної? Які типи змінних ви знаєте? Чим вони відрізняються один від одного?

3. Як увести та вивести дані під час виконання проєкту без графічного інтерфейсу користувача?





Розділ 6
§ 6.5

Домашнє завдання



Проаналізувати
§ 6.5, ст. 194-202